ISSN 1870-4069

Análisis de ataques de denegación de servicio distribuido (DDoS) en un conjunto de datos usando aprendizaje automático

Jesus Barranco Castillo, Arturo Zúñiga-López,
Carlos Avilés-Cruz, Cesar Benavides-Álvarez

Universidad Autónoma Metropolitana,
Unidad Azcapotzalco, Departamento de Electrónica,
México

{al2163054940, azl, caviles, cesarbenavides}@azc.uam.mx

Resumen. La seguridad en las redes es y seguirá siendo un tema de actualidad, pues la cantidad de ataques hacia empresas, instituciones o incluso personas, es una constante hoy en día. Gracias a las herramientas que se tienen actualmente, los atacantes ya no necesitan tener grandes conocimientos sobre redes y programación para realizar ataques muy sofisticados. En este trabajo se utilizó la base de datos que contiene ataques DDoS, creada por el Instituto Canadiense de Ciberseguridad llamada "CICDDoS2019". La base de datos fue analizada, utilizando las técnicas de red neuronal artificial (RNA) y máquinas de soporte vectorial (MSV). Los datos también fueron tratados con análisis de componentes principales (PCA) para reducir la dimensionalidad, además de reducir el poder de cómputo necesario al entrenar y predecir los datos. De las técnicas utilizadas se obtuvieron buenos resultados ya que en clasificación binarias (determinar si existe ataque) se obtiene una exactitud del 99% y de clasificación múltiple (si existe ataque y de que tipo es) se obtiene una exactitud del 96%, con la técnica de RNA, que fue la técnica que mejores resultados obtuvo.

Palabras Clave: Ataques DDoS, redes neuronal artificiales, máquinas de soporte vectorial, análisis de componentes principales.

Analysis of Distributed Denial of Service (DDoS) Attacks on a Dataset Using Machine Learning

Abstract. Network security is and will continue to be a topical issue, since the number of attacks against companies, institutions or even individuals is a constant nowadays. Thanks to the tools currently available, attackers no longer need to have great knowledge of networks and programming to carry out very sophisticated attacks. In this work, the database containing DDoS attacks created by the Canadian Cybersecurity Institute called "CICDDoS2019" was used. The database was analyzed using artificial neural network (ANN) and support vector

machines (SVM) techniques. The data were also treated with principal component analysis (PCA) to reduce dimensionality, as well as to reduce the computational power required for data processing and prediction. Good results were obtained from the techniques used, since in binary classification (determining if there is an attack) an accuracy of 99% was obtained and in multiple classification (if there is an attack and what type it is) an accuracy of 96% was obtained, with the ANN technique, which was the technique that obtained the best results.

Keywords: DDos attacks, artificial neural network, support vector machines, principal component analysis.

1. Introducción

La red es uno de los principales medios de transporte de la información en la actualidad, es por esto, que es muy propensa a sufrir ataques por parte de agentes malintencionados cuya finalidad es perjudicar a otros usuarios, en la actualidad tenemos diferentes tipos de ataques, como lo son ataque de ARP (protocolo de resolución de direcciones) spoofing, Man-In-The-Middle, denegación de servicio, denegación de servicio distribuido, etc., por lo que este trabajo está enfocado en los ataques de denegación de servicio distribuido.

Un ataque de denegación de servicio (DoS – Denial of Service), es un tipo de ataque dentro de una red, el cual consiste en hacer que un recurso o servicio no se encuentre disponible y entonces perjudica tanto al que ofrece el servicio como al que lo utiliza. Este ataque se puede volver más complejo, cuando el ataque no lo realiza un equipo sino múltiples equipos a la vez, entonces esto es conocido como ataque de denegación de servicio distribuido (DDoS). Existen varios tipos de ataques DDoS, sin embargo, el presente trabajo se enfoca en dos categorías: Los ataques DDoS basados en reflexión y los ataques DDoS basados en la explotación.

Los ataques DDos de reflexión se pueden llevar a cabo usando protocolos de capa de transporte, como lo son el protocolo de control de transmisión (TCP) con ataques MSSQL (Microsoft SQL) y SSDP (protocolo simple de detección de servicio), y el protocolo de datagramas de usuario (UDP) con ataques CharGen (protocolo generador de caracteres), NTP (protocolo de tiempo de red) y TFTP (protocolo trivial de transferencia de archivos), también hay otros tipos de ataques como son DNS (sistema de nombres de dominio), LDAP (protocolo ligero de acceso a directorios), NETBIOS (sistema de entrada salida básica de red) y SNMP (protocolo simple de gestión de redes) que utilizan los protocolos antes mencionados.

Los ataques DDoS basados en la explotación también se pueden llevar a cabo usando protocolos de capa de transporte (TCP y UDP). Los ataques de explotación basados en TCP incluyen inundación SYN (bandera que indica el inicio de conexión entre host) y los ataques basados en UDP incluyen inundación UDP y UDP-Lag.

Por otro lado, el aprendizaje automático permite desarrollar herramientas que puedan brindar apoyo a la seguridad en las redes y en los sistemas de información, ya que permite que las computadoras puedan tener un proceso de aprendizaje, identificar patrones en un amplio conjunto de datos y obtener predicciones basadas en dichos patrones. Dentro del aprendizaje automático se tiene una rama llamada aprendizaje

supervisado (AS) la cual es usada en este trabajo, en donde se conocen claramente las características (entrada de información) y etiquetas (salidas), las cuales son usadas en el proceso de entrenamiento y en la determinación de la precisión del modelo de aprendizaje.

En este trabajo se usó el conjunto de datos generado por el Instituto Canadiense de Ciberseguridad (CIC) [1] llamado "CICDDoS2019" [2], el cual contiene una taxonomía de tráfico en la red con características importantes para detectar ataques de DDoS.

2. Antecedentes

En el trabajo [3] se analizan ataques y tráfico normal en el conjunto de datos "CICDDoS2019" en donde se aplican las redes convolucionales ResNet50 y EfficientNet. Dado que dichas redes funcionan mejor con imágenes, se transforman los datos del tráfico de la red a imágenes, las cuales se introducen en esas redes para una clasificación múltiple (12 clasificaciones diferentes). Se determina como mejor solución a EfficientNet con 0.903 de precisión. En [4] propone DDosNet basada en red neuronal recurrente-autoencoder para la detección de ataques DDoS, se utiliza los datos de "CICDDoS2019" y se realiza una clasificación binaria, la red propuesta obtiene 0.99 de precisión al detectar ataques y 1.0 al detectar tráfico normal.

En [5] proponen la taxonomía de CiCDDoS2019 como un nuevo conjunto de datos con características importantes para la detección de ataques DDoS, se implementan cuatro modelos conocidos de aprendizaje automático los cuales son ID3, Random Forest , Naïve Bayes y regresión logística con precisiones de 0.78, 0.77 0.41 0.25 respectivamente. En otro trabajo [6] se utiliza el conjunto de datos "CICDDoS2019, para determinar ataques de DDoS por reflexión en cuatro diferentes categorías (Portmap, LDAP, NetBIOS y MSSQL) y donde se obtuvo las siguientes precisiones Portmap 0.9958, LDAP 0.9999 NetBios 0.9999 y MSSQL 0.9999.

En el trabajo [7], proponen las técnicas de red neuronal artificial Naïve Bayes y máquinas de soporte vectorial también utilizan la base de datos CICDDoS 2019 y se determinó a la red neuronal artificial como la técnica más precisa con 0.99 de precisión, Naïve Bayes y máquinas de soporte vectorial se obtuvieron 0.982 0.981 de precisión respectivamente.

En [8] Comparan la precisión entre Snort (un sistema de detección de intrusos) y los sistemas de detección basados en aprendizaje profundo para detectar ataques DDoS. Presentaron que la precisión de Snort, RNA, RNA apilado y CNN (red neuronal convolucional) es 0.4716, 0.9976, 0.9956, respectivamente. Sin embargo, a Snort le tomo poco tiempo en detectar los ataques.

Hou et al. [9] introduce un método para detectar tráfico DDoS mediante aprendizaje automático. Extrajeron características basadas en flujos a partir de datos de muestreo en tiempo real utilizando NetFlow. Los resultados muestran que la precisión promedio es superior al 99% y un falso positivo inferior al 0.5%.

Roopak et al. [10] proponen un modelo híbrido CNN+LSTM que tiene una mayor precisión que otros modelos de aprendizaje profundo y modelos de aprendizaje automático. Se utilizó el conjunto de datos de CICIDS2017 y la precisión de detección de flujos de ataque es del 97.16%.

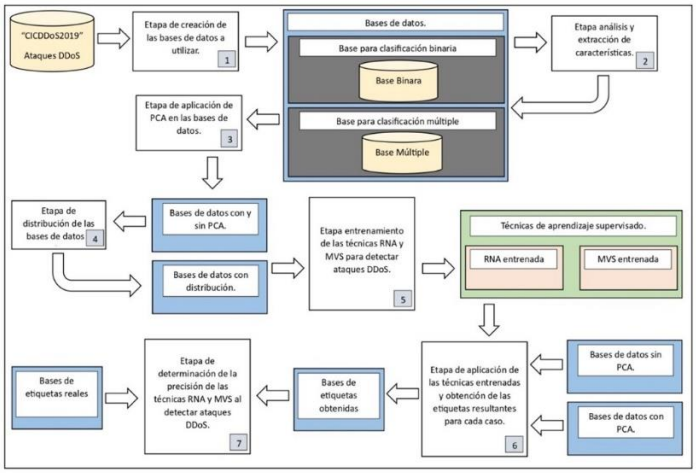

Fig. 1. Diagrama de le metodología.

En [11] proponen un sistema de defensa contra instrucciones y ataques DDoS. Utilizaron el método GRU (Gated Recurrent Units) en el sistema de detección. Se utiliza CICDDoS2019, el resultado promedio de exactitud es de 99.94% que es más alto que otros métodos, como DN, CNN, LSTM y máquinas de soporte vectorial.

3. Metodología

En la figura 1 se muestra las distintas etapas de la metodología, en donde los recuadros azules son una representación del conjunto de la base de datos a utilizar tanto para la clasificación binaria, es decir en esta etapa el sistema solamente determina si hay o no un ataque de DDoS, y para la clasificación múltiple, el sistema determina si no hay un ataque o si existe, qué tipo de ataque es el que hay de 8 posibles tipos. Algunas de las etapas importantes son:

La etapa de creación de la base de datos a utilizar tiene el objetivo de generar una nueva base de datos con los tipos de ataques y con los atributos que consideramos más relevantes, por ejemplo, se descartan los atributos IP (protocolo de internet) de origen y destino, puerto de origen y destino. Con la nueva información creada se generar varias bases de datos como son: la de la clasificación binaria donde únicamente hay dos posibles salidas (ataque o no ataque), la de clasificación múltiple en donde existe 9 posibles salidas que no haya ataque o un tipo de ataque en particular, por último, las bases de datos de se obtuvieron aplicando PCA a los datos.

En la etapa de análisis y extracción de características se extraen las características más relevantes con el fin de reducir el tiempo y el nivel de procesamiento de cómputo durante la etapa de entrenamiento. Por último, en la etapa de la precisión se aplica la métrica “label_ranking_average_presision_score” de Sklearm para determinar la precisión de las técnicas.

Dado que “CICDDoS2019” [2] se encuentra distribuida en bases de datos con ataques DDoS de diferente tipo y cada base cuenta con una pequeña porción de tráfico

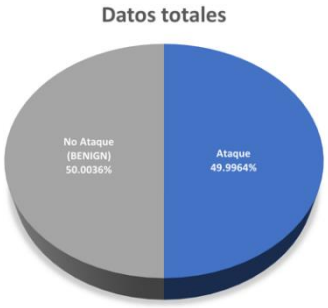

Fig. 2. Distribución de los datos para clasificación binaria.

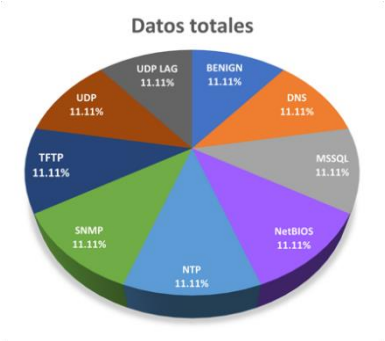

Fig. 3. Distribución de los datos para clasificación múltiple

normal, primero fueron separados los ataques del tráfico normal y posteriormente fueron unidos en una sola base de datos con una cantidad homogénea de datos para cada clasificación.

Lo anterior fue realizado para dos bases de datos, la primera base cuenta con 34,661 datos sobre tráfico normal el cual fue llamado "BENIGN", y con 34,656 datos sobre ataques DDoS (4,332 datos para cada tipo de ataque), esta base de datos fue usada para una clasificación binaria por lo que las etiquetas de los datos tipo "BENIGN" fue cambiada a 0 y las etiquetas de los datos sobre ataques fue cambiada a 1, La figura 2 muestra la distribución de la base binaria.

La segunda base de datos cuenta con 34,661 datos "BENIGN" y 34,661 datos por cada tipo de ataque (311,949 datos en total), esta base de datos fue usada para una clasificación múltiple, donde las etiquetas fueron cambiadas de la siguiente manera: BENIGN: 0, DrDoS_DNS: 1, DrDoS_MSSQL: 2, DrDoS_NetBIOS: 3, DrDoS_NTP: 4 DrDoS_SNMP:5, TFTP: 6, DrDoS_UDP: 7 y UDP-lag: 8, la Figura 3 muestra la distribución de la base múltiple.

4. Trabajo propuesto

En esta sección se muestran las redes propuestas para detectar ataques DDoS en las bases de datos definidas anteriormente.

Clasificación binaria: Se propone una RNA para clasificación binaria, la cual es usada con datos sin y con PCA, con la diferencia de que la capa de entrada es de 69 y 21 (estos atributos se eligieron porque la suma de ellos describe la mayor cantidad de varianza, aproximadamente el 98%) neuronas respectivamente. Se trata de un modelo secuencial con una capa de entrada (las neuronas son definidas por los datos con y sin PCA), cinco capas ocultas de 64, 32, 16, 8 y 4 neuronas respectivamente y una capa de salida de dos neuronas, fue usada la función de pérdida "binary_crossentropy" y el optimizador "adam". También fue usada la técnica Máquinas de soporte vectorial (MSV) con kernel "linear".

Para la clasificación múltiple: Se propone una RNA con todos los datos y con datos obtenidos de PCA donde la capa de entrada para datos sin PCA es de 69 y 21 para datos con PCA. Es un modelo secuencial con una capa de entrada (las neuronas son definidas

Valores esperados	Verdadero Positivo	Verdadero negativo	Falso Positivo	Falso Negativo
Real	Benign	Ataque	Ataque	Benign
Predicción	Benign	Ataque	Benign	Ataque
Técnica Usada	Valores Obtenidos			
RNA	10,411	10,371	5	9
RNA (PCA)	10,409	10,372	4	11
MVS	10,385	10,363	13	35
MVS (PCA)	10,385	10,362	14	35

Fig. 4. Distribución de los datos para clasificación múltiple.

Precisión de las técnicas			
Técnica	**Clasificación**	**PCA**	**Precisión.**
RNA	Binaria	No	0.9996633968070783
		Si	0.9996393537218696
	Múltiple	No	0.9632704671332124
		Si	0.9632609689349022
MVS	Binaria	No	0.9988459319099827
		Si	0.9988218888247740
	Múltiple	No	0.8387490872822809
		Si	0.7958457255136616

Fig. 5. Precisión de las técnicas RNA y MSV.

por los datos con y sin PCA), cinco capas ocultas de 1280, 640, 320, 80 y 20 neuronas respectivamente y una capa de salida de 9 neuronas.

Además, después de las capas de 1280, 640, 320 y 80 neuronas, se hizo un Dropout de 0.35, 0.30, 0.25 y 0.20 respectivamente, con el fin de evitar el sobre entrenamiento y tener mejores resultados, fue usada la función de pérdida "binary_crossentropy" y el optimizador "ADAM". También fue usada la técnica máquinas de soporte vectorial (MSV) con kernel "poly".

5. Resultados

5.1 Clasificación binaria

Como se puede observar en la Figura 4 ambas técnicas tienen buenos resultados en una clasificación binaria, además en la figura 5 se puede observar que la RNA propuesta tienen mayor precisión (Siendo estrictos) que la MSV propuesta.

Como podemos observar en la figura 4 y en la figura 5 la RNA propuesta tiene casi los mismos resultados cuando los datos ingresados están tratados con y sin PCA, por lo que surge la pregunta ¿es necesario usar PCA?, cuando los datos tienen PCA el tiempo en que tarda la red en realizar las predicciones baja de 1.111s a 0.8437s y la cantidad de parámetros a calcular se reduce de 7,270 a 4,198, por lo que es mejor usar datos con PCA en la red neuronal, además al tener menos parámetros de entrenamiento se utiliza menos memoria de la computadora.

5.2 Clasificación múltiple

Como se puede observar en la figura 6 los mejores resultados están en la técnica RNA sin aplicar PCA a los datos, mientras que los peores resultados se encuentran en

Precisión	RNA	RNA (PCA)	MVS	MVS (PCA)
Benign	99.89	99.90	99.68	99.54
DNS	86.79	86.60	79.24	62.45
MSSQL	96.51	96.99	94.32	94.69
NetBios	98.98	99.07	98.18	91.92
NTP	99.71	99.69	95.37	80.11
SNMP	88.35	88.14	73.73	73.10
TFTP	98.64	98.63	46.15	46.18
UDP	98.34	98.38	58.64	58.35
UDP-Lag	95.59	95.38	91.05	86.55

Peor Resultado Mejor Resultado

Fig. 6. Comparativa de las precisiones de las técnicas RNA y MVS.

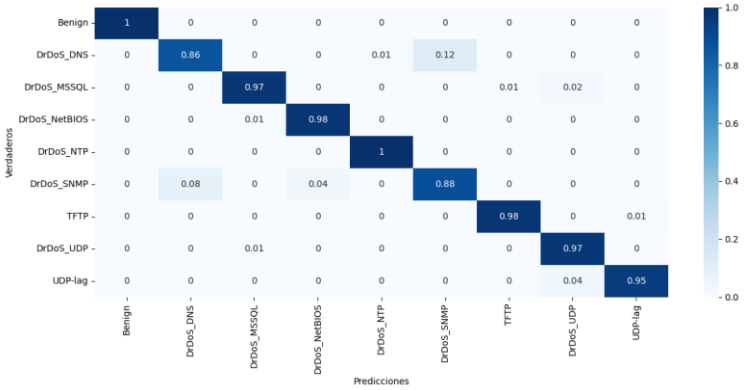

Fig. 7. Matriz de confusión para la técnica RNA.

la técnica MSV, esto se reafirma en la figura 5 donde claramente se observa como RNA es más precisa. Por lo tanto, usar la técnica RNA es mejor opción, sin embargo, vale la pena saber si es necesario usar PCA en los datos o no.

Como se puede observar en la figura 5 la precisión de RNA sin PCA es de 0.96327, mientras que con PCA es de 0.96326, además, los parámetros a calcular sin PCA son 1,142,112, mientras que con PCA son 1,081,952, por lo que hay 60,160 parámetros de diferencia, finalmente el tiempo de predicción sin PCA es de 11.2350 s, mientras que con PCA es de 10.1252 s.

Dados estos resultados, se puede determinar a la técnica RNA como mejor técnica en cuanto a precisión usando datos sin PCA, sin embargo, la diferencia de precisiones cuando se usan datos con y sin PCA es mínima, además, al usar PCA se reduce el tiempo de predicción y los parámetros a calcular, por lo que usar esta red con datos PCA es lo mejor. La figura 7 muestra la matriz de confusión normalizada en clasificación múltiple para la RNA propuesta usando los datos del análisis de PCA.

6. Conclusiones

El Instituto Canadiense de Ciberseguridad generó un conjunto de datos sobre ataques DDoS llamado “CICDDoS2019” [2], el cual cuenta con una correcta taxonomía de tráfico en cuanto a características se refiere pues los resultados muestran que es posible

realizar clasificaciones entre ataques y tráfico normal, además cuenta con una gran cantidad de datos para cada tipo de ataque, sin embargo, la cantidad de datos sobre tráfico benigno es muy poca, por lo que sería necesario tener más tráfico de este tipo para un mejor entrenamiento de la red neuronal. Sin embargo, estos datos fueron suficientes para el entrenamiento de las técnicas red neuronal artificial como máquinas de soporte vectorial. En la clasificación binaria demostraron ser eficientes al diferenciar entre tráfico benigno y tráfico con ataques, pues las dos obtuvieron 99% de precisión en sus predicciones. En la clasificación múltiple la RNA propuesta también demostró ser eficiente, pues obtuvo 96% de precisión al clasificar tráfico normal y los 8 tipos de ataques más, siendo estos ataques DNS, MSSQL, NetBIOS, NTP, SNMP, TFTP, UDP y UDP-LAG, por otro lado, la máquina de soporte de vectorial propuesta obtuvo 83% de precisión para las mismas clasificaciones.

En la clasificación binaria se determinó que es necesario aplicar PCA (con 21 componentes) a los datos, pues, aunque las precisiones son casi iguales, para datos sin PCA y con PCA respectivamente, al usar PCA se reducen los parámetros a calcular durante el entrenamiento y se obtiene un tiempo menor al realizar las predicciones. Por otro lado, en la clasificación múltiple se determinó que también es necesario aplicar PCA (con 21 componentes) a los datos, ya que la diferencia de precisiones es mínima, pero al usar PCA se reducen los parámetros a calcular durante el entrenamiento y se obtiene un tiempo menor al realizar las predicciones.

Finalmente, la RNA propuesta muestra ser eficiente en clasificaciones tanto binarias como múltiples, por lo que es posible aplicarla en un entorno real para detección de ataques DDoS.

Referencias

1. Canadian Institute for Cybersecurity: Canadian Institute for Cybersecurity, Available: https://www.unb.ca/cic/ (2022)
2. Canadian Institute for Cybersecurity: DDoS Evaluation Dataset (CIC-DDoS´19), (2019). https://www.unb.ca/cic/datasets/ddos-2019.html (2024)
3. Satpathy, S.P., Mohanty, S., Kumar, R.: A lightweight dos and ddos attack detection mechanism-based on deep learning. In: 5th International Conference on Computational Intelligence and Networks. pp. 1–6 (2022). DOI: 10.1109/CINE56307.2022.10037402.
4. Elsayed, M.S., Le-Khac, N.A., Dev, S., Jurcut, A.D.: Ddosnet: A deep-learning model for detecting network attacks. In: IEEE 21st International Symposium on A World of Wireless, Mobile and Multimedia Networks, pp. 391–396 (2020). DOI: 10.1109/WoWMoM49955.2020.00072.
5. Sharafaldin, I., Lashkari, A.H., Hakak, S., Ghorbani, A.A.: Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In: IEEE. International Carnahan Conference on Security Technology, pp. 1–8 (2019). DOI: 10.1109/CCST.2019.8888419.
6. Ahuja, V., Kotkar, M., Bhongade, R., Kshirsagar, D.: Reflection based distributed denial of service attack detection system. In: 6th International Conference on Computing, Communication, Control and Automation, pp. 1–5 (2022). DOI: 10.1109/ICCUBEA54992.2022.10011055.
7. Castillo, A., Lineses, A.B., Go, B., Labanan, R., Octaviano, M.: Trojan Malware Detection using ANN, Naïve Bayes and SVM Machine Learning Algorithms. In: IEEE 2nd

International Conference in Information and Computing Research, pp. 72–76 (2022). DOI: 10.1109/iCORE58172.2022.00033.
8. Chockwanich, N., Visoottiviseth, V.: Intrusion detection by deep learning with tensorflow. In: 21st international conference on advanced communication technology, pp. 654–659, (2019). DOI: 10.23919/ICACT.2019.8701969.
9. Hou, J., Fu, P., Cao, Z., Xu, A.: Machine learning based DDos detection through NetFlow analysis. In: MILCOM´18 IEEE military communications conference, pp. 1–6 (2018). DOI: 10.1109/MILCOM.2018.8599738.
10. Roopak, M., Tian, G.Y., Chambers, J.: Deep learning models for cyber security in IoT networks. In: IEEE 9th annual computing and communication workshop and conference, pp. 0452–0457 (2019). DOI: 10.1109/CCWC.2019.8666588.
11. Assis, M.V., Carvalho, L.F., Lloret, J., Proença, Jr, M.L.: A GRU deep learning system against attacks in software defined networks. Journal of Network and Computer Applications, vol. 177, pp. 102942 (2021). DOI: 10.1016/j.jnca.2020.102942.